

CLAIMS

What is claimed is:

1. A method of determining the flow of a data object in a software architecture using queues to organize the transfer of data from one processing object to another, comprising the steps of:

storing queue identifiers in a path object;

receiving and processing a data object in a first of said processing objects;

identifying a queue corresponding to a second of said processing objects responsively to an indicator corresponding to said data object;

placing said data object in a queue identified in said step of identifying.

2. A method as in claim 1, wherein said step of identifying includes determining a result of said step processing.

3. A method as in claim 2, wherein said step of identifying includes determining a result of said step processing and said result corresponding to said queue.

4. A method for determining the flow of data in a software architecture in which queues are used to

organize the transfer of data from one process to another process, comprising the steps of:

performing a process on a data part of a first data object, by a first processing object;

identifying a first queue to which said first data object is to be transferred from a indicator part of said first data object;

modifying said indicator part of said first data object to produce a second data object;

performing said process on said second data object;

identifying a second queue to which said second data object is to be transferred.

5. A method as in claim 4, further comprising determining a result of said step of performing, said step of identifying including identifying said second queue responsively to said step of determining.

6. A pipeline software architecture in which data objects are transferred from a first processing object to a selected one of second and third processing objects by queuing the data objects in a queue of said selected one, comprising:

6 a definition of a path object corresponding to
7 each of said data objects;

8 at least one of said path objects containing an
9 indicator of at least one of said second and third
10 processing object;

11 said first processing object defining a process a
12 result of which is to insure that a first data object
13 processed by said first processing object is placed in a
14 queue of said at least one of said second and third
15 processing objects responsively to one of said path objects
16 corresponding to said first data object.

1 7. An architecture as in claim 6, wherein said
2 process includes the generation of an indication of a
3 result of a subprocess of said first processing object and
4 said first data object processed by said first processing
5 object is placed in said queue of said at least one of said
6 second and third processing objects responsively to one of
7 said path objects corresponding to said first data object
8 and responsively to said indication.
9